IBM Systems & Technology Group

Blue Gene/L Programming and Run-Time Environment

Peter Bergner
IBM Rochester

Outline

- Programming Environment
  - Differences from Linux/PPC
    - Unsupported syscalls
    - Syscalls with limitations
- Run-Time Libraries

Programming Environment

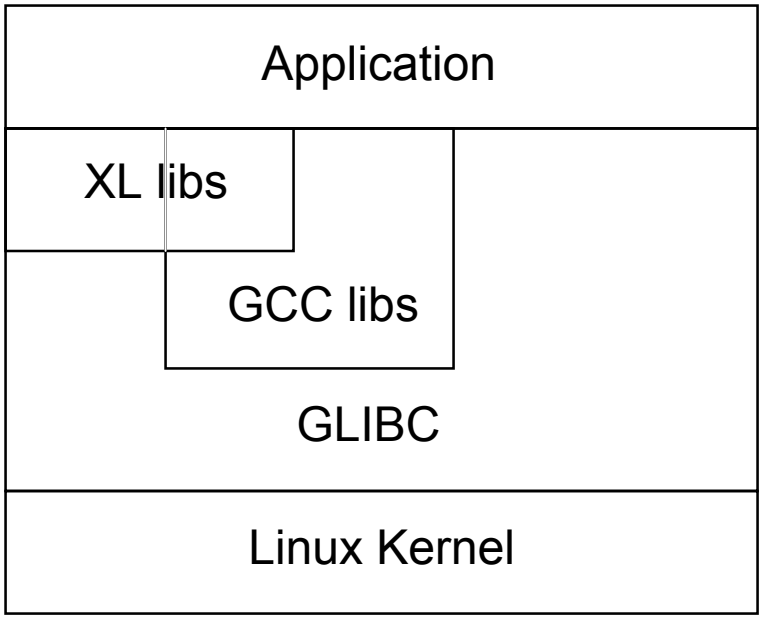
- Cross Compilation Environment
  - Front End Node running SUSE SLES9 Linux/PPC64
  - powerpc-linux-gnu -> powerpc-blrts-gnu
  - GNU toolchain for Blue Gene/L
  - IBM XL cross compilers for Blue Gene/L

Programming Environment cont.

- Similar programming model to Linux/PPC
- Differences from Linux/PPC:
  - No stdin
  - No asynchronous I/O
  - No dynamic linking
  - No demand paging / swap
    - virtual address space is mapped 1-1 with physical memory
  - No read only memory
    - due to CNK design decision
    - no SIGSEGV writing to a “const char *p”

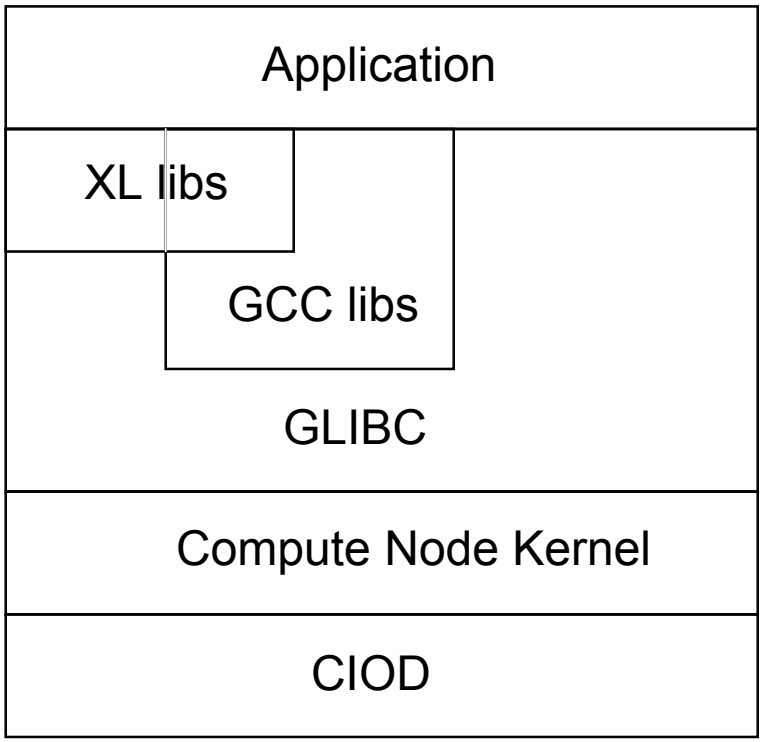
Programming Environment cont.

- Linux Software Stack



Programming Environment cont.

- Blue Gene/L Software Stack



Programming Environment cont.

- GNU Blue Gene/L toolchain
  - gcc, g++ and g77 v3.2
  - binutils (as, ld, etc.) v2.13
  - GLIBC v2.2.5
  - Blue Gene/L support supplied via patches
    - Customer applies the patches and builds the toolchain
    - IBM supplies scripts to download, patch and build everything

Programming Environment cont.

- IBM XL Compilers
  - Install IBM XLC V7.0 / XLF V9.1 compilers for SUSE SLES9 Linux/PPC64
  - Install Blue Gene/L add-on which adds Blue Gene/L versions of the XL run-time libs, compile scripts and config files.
    - Requires the GNU Blue Gene/L toolchain.
  - End result: Working Linux and Blue Gene/L compilers
    - Linux: xlc, xlC, xlf, xlf90, etc.
    - BG/L: blrts_xlc, blrts_xlC, blrts_xlf, blrts_xlf90, etc.

Programming Environment cont.

- IBM XL Compiler Options
  - -qarch=440 –qtune=440
    - Normal PowerPC FP code
  - -qarch=440d –qtune=440
    - Double Hummer FP code
  - -qhot=simd
    - Double Hummer FP code generated by TPO
  - -qipa
    - Interprocedural Analysis
  - -O4/-O5
    - Implicitly enable –qhot=simd –qipa
  - -qarch=auto, -qtune=auto, -qcache=auto
    - Disabled on Blue Gene/L

Unsupported Syscalls

- `fork, exec, clone, getppid, wait, waitpid`
- `mmap, mlock, madvise, mremap, msysnc, mprotect`
- `sigaction, sigprocmask, sigpending, sigsuspend, sigaltstack` (no POSIX signal handling)
  - We do support ANCI C signals.
- `capget, capset, getpriority, ioctl, ioperm, ipc, nice, prctl, ptrace`
- `chroot, mount`

Supported Syscalls With Limitations

- `kill(getpid(), signum)`
  - You can only send signals to yourself.
- `setitimer()`
  - You are allowed only one active timer.

Programming Environment cont.

- How to differentiate between AIX, Linux, Blue Gene/L in your code?

```
#if defined(__aix__)
    <aix code here>
#elif defined(__linux__)
    <linux code here>
#endif
```

Programming Environment cont.

- How to differentiate between AIX, Linux, Blue Gene/L in your code?

```
#if defined(__aix__)
    <aix code here>
#elif defined(__linux__)
    <linux code here>
#elif defined(__blrts__)
    <Blue Gene/L code here>
#endif
```

Programming Environment cont.

- How to differentiate between AIX, Linux, Blue Gene/L in your code?

```
#if defined(__aix__)
    <aix code here>
#elif defined(__linux__) || defined(__blrts__)
    <common linux and Blue Gene/L code here>
#endif
```

Run-Time Libraries

- GNU Run-Time Libraries
  - GCC libraries
    - GNU Standard C++ library (libstdc++.a)
    - GCC low-level run-time library (libgcc.a)
    - G77 run-time library (libg2c.a)
  - GLIBC libraries
    - GNU C library (libc.a)
    - Math library (libm.a)
    - IEEE floating point library (libieee.a)
    - G++ run-time library (libg.a)
    - Cryptography library (libcrypt.a)
    - NSS/Resolve libraries (libnss_dns.a, libnss_files.a, libresolv.a)

Run-Time Libraries

- IBM XL Run-Time Libraries
  - IBM C++ library (libibmc++.a)
    - Very light wrapper to libstdc++.a
  - IBM XLF run-time library (libxlf90.a)
  - IBM XL low-level run-time library (libxl.a)
  - IBM XL optimized intrinsic library (libxlopt.a)
    - Vector intrinsic functions
    - BLASS routines
  - IBM XL MASSV library (libmassv.a)
    - Vector intrinsic functions
  - IBM XL Open MP compatibility library (libxlomp_ser.a)

Questions?

Answers?